

# イノベーションの分散化とプラットフォーム・デザイン

著者	中田 善啓
雑誌名	甲南経営研究
巻	53
号	2
ページ	1-22
発行年	2012-09-25
URL	<a href="http://doi.org/10.14990/00002079">http://doi.org/10.14990/00002079</a>

# イノベーションの分散化と プラットフォーム・デザイン

中 田 善 啓

甲南経営研究 第53巻 第2号 抜刷

平成24年9月

# イノベーションの分散化と プラットフォーム・デザイン

中 田 善 啓

## は じ め に

経営学の分野では個々の企業が分析単位であった。しかし、個々の組織は、分析の主要な単位とはいえない状況になりつつある。1980年代以降、ビジネス・エコシステム（以下エコシステムという）における参加者によるイノベーションが創発している。エコシステムを構成するモジュールそれぞれが独立しているので、モジュールがイノベーションをスピーディーに創発することができる。さらには、オープン・プラットフォームに自由に参加する個人や組織がイノベーションの担い手となる。

伝統的にイノベーションの担い手とされてきた製造業者（メーカー）によるイノベーションの1つの方法にすぎない。それ以外に、エコシステムの参加者、たとえば、ユーザー主導のイノベーションやオープン・コラボレーションによるイノベーションがある。これらはイノベーションを創発する新しいガバナンスである。これによって、イノベーションが分散化した。これらの自律的な多数の企業、個人、コミュニティが技術的システムの進化に担い手になっている。

本稿はエコシステムにおけるイノベーションの分散化を考察する。第1節はイノベーションの分散化の要因として、モジュラー・アーキテクチャにつ

イノベーションの分散化とプラットフォーム・デザイン（中田善啓）

いて考える。モジュラー・アーキテクチャが産業の階層システムからエコシステムへと進化し、情報技術の進歩がエコシステムのイノベーションを加速したことを明らかにする。さらに、エコシステムで外部のイノベーターの参加が重要であることを示す。

第2節はエコシステムの調整者としてマルチサイド・プラットフォームが効率的であることを明らかにする。同時にプラットフォームのオープン化がイノベーションを促進することを見ていく。第3節はイノベーション・コミュニティのケースをとりあげる。それらは、Tシャツのプラットフォームとしてスレッドレスコム（Threadless.com）、科学的問題解決のプラットフォームとしてイノセンティブコム（InnoCentive.com）、ソフトウェアのプラットフォームとして、トップコーダー（TopCoder）である。第3節はイノベーション・コミュニティの特色をとりあげ、コミュニティへの参加者の誘因を考える。

## 1. モジュラー・アーキテクチャ

### 1.1 モジュール化

イノベーションの分散化の大きな要因の1つは、基本的な製品またはプロセスがモジュール化しているためである。モジュラー・アーキテクチャはシステム全体を相互作用が少ないか、まったくないようなモジュールの集合に分解して、それぞれのモジュールがどのように相互に機能するかを特定化するデザイン・ルール（標準）を作ることになる。したがって、モジュラー・アーキテクチャは、モジュールといわれるサブシステム内では相互依存関係が強く、サブシステム間では独立しているような要素に分解し、それらを全体システムとして再統合するデザインルールである。モジュールはそれぞれ独立しているので、モジュール内の変化はそれ以外のモジュールには影響を及ぼさない。これを情報隔離（information hiding）という<sup>(1)</sup>。情報隔離がある

ために、システムの環境の小さい変化が全体のシステムに影響を及ぼし、システム全体が機能不全になるリスクが減少するので システム全体が環境に適応しやすく、パフォーマンスが向上するように進化することができる。

ユニークな知識は価値のある製品またはプロセスをつくるために必要であるので、その知識のコントロールが利益の源泉となる。その知識を他者が利用できない場合に、それは財産権となる。人間の認知能力は限られているので、知識は専門化される。複雑な製品またはプロセスのデザインの主要な課題は、タスクとそれに関連する知識を分解して、それに関心を持つ組織や個人が問題解決できるように、知識をコミュニケーションし、共有することである。システムのアーキテクチャがデザインルールとなって下位問題をコントロールする。

システムをモジュール化できなければ (インテグラル・アーキテクチャ)、すべてのデザイナーは他のデザイナーが何をしているかについての知識と論理を共有しなければならない。このケースでは情報が完全に共有されているので、システム・アーキテクチャは必要ではない。しかし、モジュラー・アーキテクチャは、下位問題を独立のモジュールに分解するので、あるモジュール間のインターフェイスを共有しなければならない。各モジュール・デザイナーはデザインルールにアクセスする必要があるが、デザインルールを守る限り、多数のモジュールはシステム全体として機能する。

モジュール化は自然発生的に生まれるのではなく、システム・デザイナーが人工的に作り出した技術システムである。どのようなモジュール化の構造を選択するかは物理的法則や知識の制約を受けるが、システム全体がある程度分解され、それぞれのサブシステム (モジュール) で分業が行われていることが多い。その意味では多くの複雑な技術的システムは多かれ少なかれモ

---

(1) 情報の隔離については、Baldwin [2008], 中田 [2011] を参照。

イノベーションの分散化とプラットフォーム・デザイン（中田善啓）

ジュール化されている。しかし、同じシステムであっても、組織によってモジュールの境界は異なっている。

モジュール化は企業間の競争に影響を与える。モジュール間の取引が長期継続的ではなく、競争メカニズムが機能するので、モジュールの代替性が働き、製品やシステムの模倣が容易になる。一方ではイノベーションが加速されるので、模倣を上回る可能性がある。モジュール化はモジュール間の相互依存性を排除するので、モジュールの複雑性が減少する。その結果、デザインが単純になるので、模倣が容易になる。しかし、モジュール間の相互依存関係がないので、モジュールのデザイナーはモジュールレベルの実験に集中して、他のモジュールとは関係なく、問題解決に集中できる。したがって、モジュール化は模倣と非代替性のトレードオフの性格をもつ。

## 1.2 情報技術の進歩

モジュラー・アーキテクチャはイノベーションの分散化の必要条件であるが、十分条件ではない。しかし、情報技術がモジュラー・アーキテクチャをサポートすると、イノベーションが加速し分散する。組織デザインは時間的、空間的、技術的な制約を受けるが、特にコミュニケーションと情報処理の技術の影響を受ける。これらイノベーションにより、組織内の適応的調整をリアルタイムに行うことができるようになった。組織はこれらを使って、大量のデータを低コストで、スピーディーに処理することが可能であるので、多数の技術、部品、製品の新しいアイデアを実験することができる。<sup>(2)</sup>

情報処理費用が低下すると、物理的なプロトタイプを作らなくても、デザイン変更の影響をシミュレートできるので、新しいデザインの評価がスピーディーに行うことができる。同時に、デザイン全体を変更しないで、モジュ-

---

(2) Baldwin and von Hippel [2010], 中田 [2010].

ルのオプションを実行することができる。情報処理やコミュニケーションの費用が低下すると、デザイナーやエンジニアは技術的なシステムのアーキテクチャをモジュールが全体に適合するようにコード化して、個々のモジュールとアーキテクチャで実験やシミュレートすることができる。情報処理費用が高いと、これらは実行可能ではない。

このように、20世紀後半における情報技術革新による情報処理とコミュニケーション費用の低下はモジュラー・アーキテクチャの進化を促した。これらの要因はイノベーションの分散化を促進したが、イノベーションは意図しない、予想外の結果である。<sup>(3)</sup> IBM が IBM/360 にモジュラー・アーキテクチャを採用した。<sup>(4)</sup> IBM は顧客満足と規模の経済を高め、複雑な製造を単純化しようとして、デザインをモジュール化した。これによってディスク・ドライブのような周辺機器デバイスが互換的になり、製造業者間の競争がイノベーションを分散化した。

また、モジュール化はコンピュータの階層的構造からエコシステム化へと移行させた。IBM は PC についても、費用を減らし、新製品のアピールを高めるためにエコシステム化を構築した。IBM の経営陣はすべての PC システムの販売から利益をあげ、市場の成長をコントロールして、独占的地位を確保できると考えていた。IBM のマネージャーは、PC をモジュール化して、そのハードウェア部品およびソフトウェアのほとんどを外注化したが、重要なコードである BIOS (Basic Input Output System) を所有し、クローズドにした。

しかし、コンパックとフェニックス・テクノロジーは互換的な BIOS を開発したので、IBM クローンが登場した。結果的に、IBM は外部イノベーターに PC をオープン化し、エコシステム化が進化した。IBM は2004年に PC

---

(3) Baldwin [2012].

(4) Baldwin and Clark [2012]. 1.

イノベーションの分散化とプラットフォーム・デザイン（中田善啓）

市場から撤退したが、結果的にイノベーションを分散化を促進した。

### 1.3 ジョイの法則

イノベーションはある領域での創造的な問題解決の結果である。創造性はある領域で新しい、有益なアイデアを考え出すことである。<sup>(5)</sup> 組織は組織内の創造的なアイデアを製品やサービスに埋め込まなければならない。創造的な問題解決者の思考と行動の習慣が多様であるので、特定の問題のベストな解決者は、世界中に分散している。サンマイクロシステムズの元チーフサイエンティストのジョイ（B. Joy）によれば、優秀者な人材は他人のために働いているわけではなく、イノベーションが自組織の外部で起こることを想定する必要がある。<sup>(6)</sup> このジョイの法則によれば、多くの関連する知識が企業外に存在するので、イノベーションの重要な課題は外部の知識にアクセスし、とりこむことができるかどうかである。

ジョイの法則は、個人ないしは企業がイノベーションの成果を所有するようなクロズド・システムが、知識集約的なタスクを完成するには効率的ではないことをいっている。このような知識の分散化に加えて、フォン・ヒッペル（E. von Hippel）は製造業者（メーカー）がイノベーションを主導するよりも、ユーザーがイノベーションを組織化するユーザー・イノベーションが重要であることを実証的に明らかにして、イノベーションの分散化を指摘した。<sup>(7)</sup>

そこで、外部のイノベーションを取り込む戦略が必要になる。それぞれの創造的な問題解決者は多様な環境のもとで独自にテーマを選択している。新

---

(5) Amabile and Kramer [2011] p. 49-51.

(6) Lakhani and Pnetta [2007], Baldwin [2012]. ジョイは UNIX を改良すると同時に、インターネットの普及に貢献した。

(7) von Hippel [2005].



規事業に取り組むような解決者もいれば、自分で問題を選んで、その解決に専念したい解決者もいる。また、同じテーマをもつ個人がコミュニティを構築したいかもしれない。重要なことは、多様な個人がそのスキルを特定の問題に応用して、その解決に集中できるようなガバナンスを構築することである。

問題解決者の誘因は多様で、知的好奇心、金銭、名声、またはこれらがミックスしているかもしれない。同時に、自分のためないしは他者のために問題解決に取り組むのかもしれない。いずれにせよ、創造的な問題解決は標準的な雇用契約、販売契約、供給契約と異なる点は、問題解決者が問題を自己選択し、その成果ないしはアウトプットをコントロールすることである。後者は成果ないしはアウトプットについて問題解決者に財産権を与えることである。ただし、問題解決者は自分の成果から利益を得ることもできるし、フリーで公開することもできる。

エコシステムはイノベーション化を促進する。エコシステムは多様なモジュールから構成されているので、自律性を重視する創造的な問題解決者を誘引でき、しかもそのアイデアに対してコントロール権（所有権）を与えることができる。階層システムでは自律的な問題解決者の創造力と独創力を喪失させることが多い。一方、エコシステムは問題解決者の創造性と独創力を発揮できるように、誘因を与えることができると同時に、モジュールが代替可能であるので問題解決者間で競争的にするようにデザインできる。

## 2. エコシステム

### 2.1 プラットフォーム

エコシステムは創造的な問題解決者の努力を引き出すようにデザインされなければならない。技術的システムは要素（部品）からなり、要素はモジュールないしはデザインルールである。モジュールは最終製品の部品として存在

イノベーションの分散化とプラットフォーム・デザイン（中田善啓）

し、デザインルールはモジュールがどのように設計されるかを決定して、各モジュール間の機能とその制約を規定する。<sup>(8)</sup> デザインルールをプラットフォーム・アーキテクチャという。デザインルールは標準（standard）であって、モジュールがどのように活動するかを特定化する。このようなルールがなければ、システムは機能しない。

モジュールとデザインルールからなる技術システムはコアの要素とオプションの要素からなり、それらの要素は補完的な関係にある。しかし、補完性の程度は異なる。<sup>(9)</sup> コアの要素がなければ、システムは機能しないが、オプションの要素がなくても機能する。OS と MPU は同じレベルで相互に補完的であって、どちらが欠けても PC は機能しない。ウィンドウズとメディアプレイヤーは補完的ではあるが、その程度は同じではない。メディアプレイヤーはウィンドウズがなければ機能しないが、ウィンドウズはメディアプレイヤーがなくても機能する。

コアとなるモジュールの所有者はシステム全体から得られる利益の配分について、オプションの要素の所有者に対して交渉力を持つ。<sup>(10)</sup> コアとなるモジュールが欠けると、オプションの要素が機能しないからである。そこで、コアのモジュールがシステムからの利益の配分を優先的に得て、その残余の利益をオプションのモジュールが得る。

オプションの要素について創造的な問題解決者の利益はコアの要素の所有者の数が多いほど小さくなるので、外部の自律的なイノベーターの参加する誘因は小さくなる。コアの要素の所有権を集中すれば、それを分割する場合よりも外部のイノベーターによる投資を増やすことになる。そこで、システ

---

(8) Baldwin [2012], Baldwin and Woodard [2011].

(9) Chen and Nalebuff [2006].

(10) 所有権と利益の配分については Hart and Moor [1990], Baker, Gibbons and Murphy [2002] の所有権モデルを参照。

ムは技術的なシステムの要素集合を2つのグループに分解するのが効率的になる。第1はコアとなる要素の集合からなるプラットフォームであり、第2はそのプラットフォームに補完的なオプションの要素の所有者である外部補完業者の集合である。<sup>(11)</sup>多数のスポンサーがプラットフォームを所有するよりも、1社が所有する方が効率的である。

コアとなる要素の集合がプラットフォームであるので、プラットフォームは長期的に安定的な要素からなる。これに対して、オプションとなる要素はプラットフォームに補完的であるが、代替可能であるので、変動的である。PCではOSとMPUがプラットフォームであり、周辺ハードウェアやアプリケーション・ソフトウェアはオプションの要素である。インターフェイスはオプションとなるモジュールの境界を規定すると同時に、モジュール間の相互依存関係を希薄にするので、プラットフォームの重要な要素である。

プラットフォームは要素（モジュールとデザインルール）に分解できるが、前述のようにこれらの要素を分割して所有することは、一般的には効率的ではない。そこでプラットフォーム・スポンサーはこれらの要素を統一的に所有する。コアの要素を分割したケースは、IBM PCである。IBMはMPUをインテルに、OSをマイクロソフトに分割した。IBMはデザインや技術的理由からコアの要素を分割所有させたのではなく、早急にPCを開発する戦略をとったためである。結果的にはIBMにとってこの分割は失敗であった。

さて、プラットフォームはプラットフォームのタイプには次の3つに大別できる。<sup>(12)</sup>

- (1) 企業内プラットフォーム
- (2) サプライチェーン・プラットフォーム
- (3) マルチサイド・プラットフォーム (multi-sided platform)

---

(11) Baldwin [2012], Baldwin and Woodard [2011], p. 25.

(12) Gawer [2009], 中田 [2011].

これらのプラットフォームは再利用と多様化を容易にできるように、モジュール化する。さらに、構成要素の間で相互作用を媒介するインターフェイスを指定し、要素内の行動はモジュール間で独立している。(1)と(2)は従来から製品プラットフォームとよばれ、特定の企業や企業グループ内でクローズド・システムで利用されている。マルチサイド・プラットフォームは複数の異なる顧客間の取引を促進し、<sup>(13)</sup> 程度の差があるが、オープン・システムである。

企業内とサプライチェーン内のプラットフォームが企業内とグループ内で範囲ないしは規模の経済性を提供するように、マルチサイド・プラットフォームもこれらの経済性を顧客に提供する。マルチサイド・プラットフォームは複数の異なる顧客間の取引を促進するので、顧客間の調整を行う必要がある。アーキテクチャの固定点がこの調整を行う。たとえば、それはピザの支払い処理システムであり、インターネット場合にはTCP/IPのプロトコルのようなシステム間の互換性に関する技術的標準、ショッピングセンターでは建物のロケーションである。

従来の組織デザインは個々の企業に焦点を当て、類似した製品を生産、販売する企業の競争の優位性が分析の主な対象であった。エコシステムに参加している企業の製品価値は差異がみられるし、各企業は製品や効率性をアピールする。エコシステムの市場に参入や退出は頻繁に行われる。しかし、エコシステムで従来の個々の企業間の競争では見られないようなコミュニティ化がおきている。コミュニティは問題解決者が自律的に参加し、問題を自己選択し、問題解決に関する知識を共有するオープン・システムである。エコシステムはそのメンバーが競争する一方で、大規模なシステムに進化していく。エコシステムは新規のメンバーの参入や退出があるので、開発についてメンバー間の組み合わせが変化して、無数の開発機会が生まれる。

---

(13) Hagiu and Wright [2011], 中田 [2012].

たとえば、フェイスブックを考えてみよう。フェイスブックはSNSのエコシステムにおけるプラットフォームである。フェイスブックは社会的ネットワークのウェブサイトで、ユーザーがコンテンツを供給し、広告から収益をあげている。これだけをみれば、フェイスブックは伝統的な広告支援ビジネスと変わらないが、その活動は伝統的な企業中心の組織デザインとは大きく異なっている。

フェイスブックは、World Wide Web プロトコル、有線ないしは無線のインターネットと物理的下位基盤に支えられている。また、PC やスマートフォンはオープンソースのコードベースに依存して、ウェブサイトでメンバー間のコミュニケーションが管理されている。このコードベースはフリーのモジュールである。フェイスブックはエコシステムから創発する分散的なイノベーションからの要素を再結合して、ソーシャル・ネットワーキングという新しい分野を創出した。

エコシステム内のイノベーションの分散化はハイテクテクノロジー産業、情報集約的産業、アパレル産業やバイオテクノロジー産業にもみられる。繊維産業ではゴアアソシエートはプラットフォームとなって、防水ファブリック（モジュール）を、アパレル、アウトドアのシューズのメーカー利用についてのルールつきでライセンスしている<sup>(14)</sup>。

## 2.2 オープンソース・プラットフォーム

通常1つの企業がプラットフォームを所有して（スポンサーとなって）、補完業者（オプションのモジュール）からライセンス料を得る。そのために、オプションの要素がイノベーションを利用するのに制約がある。そこで、外部イノベーターを誘引するにはスポンサーがプラットフォームフリーでオー

---

(14) Boudreau and Lakhani [2009], 中田 [2009], 第6章を参照。

イノベーションの分散化とプラットフォーム・デザイン（中田善啓）

ブン化するか、所有権を設定しない方が効率的である。外部のイノベーターにとって魅力的な環境は、開発者がフリーでプラットフォームのコアのモジュールを利用できることである。そうすると、オプションのモジュールはプラットフォーム・スポンサーと利益を分配する必要はない。外部イノベーターはコアのモジュールをフリーで利用できれば、イノベーションへの投資を増やすであろう。

リナックスの創始者のトーバルズ（L. Torvalds）は OS をオープン・ソース化して、その開発を加速化した。リナックスのカーネルのようなコアのモジュール化をフリーにすると、外部のイノベーターが開発する誘因が大きくなる。オープン化はイノベーションを独占的に所有するクローズド・システムとはまったく異なるガバナンスである。インターネットや WWW が所有権のあるネットワークよりも大きく成長した理由は、多くのデベロッパーが開発にフリーで参加できたことである。フリーのネットワーク・プロトコルと低価格のモデム（モジュール）からなるシステムは所有権のあるネットワークを消滅させた。

最近、フリーのコア・システムを供給する明確な目的のために、新しいガバナンスが創発した。第 1 はオープンソース・コミュニティであり、その例はアパッチ財団、リナックス財団等の開発コミュニティである。第 2 は、Internet Engineering Task Force や World Wide Web Consortium のように標準化を促進する組織である。これらのマルチサイド・プラットフォームは排他的にルールを設定する。プラットフォームのメンバーはこれらの組織を所有し、オプションの要素を開発する。

さらに、IBM、アップル、オラクルのような知的財産権を所有する企業が、OSS（オープンソース・ソフトウェア）コミュニティを受け入れて、社員にコミュニティに参加させ、著作権のあるソフトウェアを OSS コミュニティに寄付したり、その成果を戦略的製品やサービスに組み込んだりしている。

OSS コミュニティは、分権的な問題解決、自主的な参加、調整とコラボレーションの自己組織化、知識の無料公開によって、コミュニティと利益を同時に追求するハイブリッドのガバナンスである。このような OSS コミュニティはアパレル、バイオテクノロジー、ミュージック、エンターテインメントの領域で拡大している。

### 3. イノベーション・コミュニティのビジネスモデル

#### 3.1 スレッドレス コムのビジネスモデル

外部のイノベーションを取り込むガバナンスのビジネスモデルが進化している。以下で、3つのケースをとりあげて、その特色を明らかにしよう。

まず、スレッドレスコムはTシャツのプラットフォーム企業で、コミュニティ・ベースの分散的イノベーション<sup>(15)</sup>を利用する企業である。スレッドレスコムはデザイナー、メーカー、ユーザーを顧客とするプラットフォームである。アパレルなどのファッション企業は重要な課題に直面する。それは、デザイナーを適切な時期にヒット商品をデザインし、生産サイクルと需要サイクルを同期化することである。スレッドレスコムは世界中の顧客がイノベーション、新製品開発、販売予測、マーケティングを行うようなシステムを構築している。スレッドコムはデザインを誰からもでも受け入れ、応募デザインをコミュニティ・ベースで選択するプラットフォームを構築して、アパレル特有の問題を克服している。

スレッドレスコムのビジネスモデルはグラフィック・デザイナーやアマチュアのデザイナーが新しいTシャツのデザインを応募して、顧客が5段階評価の投票をし、購入意思を明らかにする。これによって、販売予測がかなり正確になる。スレッドレスコムこの情報をもとに毎週6から10種類の新し

---

(15) Lakhani and Panetta [2007]. スレッドコムについては <http://www.threadless.com/retail/> 参照。

イノベーションの分散化とプラットフォーム・デザイン（中田善啓）

いTシャツの生産を行う。コンテストでトップのデザイナーはウェブサイト  
で公表され、2,500ドルを賞金として受け取り、Tシャツにデザイナーの名  
前が印刷される。コミュニティ・メンバーは応募デザインを批評し、デザイ  
ナーがアイデアを改善するようにフィードバックする。オンライン・コミュ  
ニティに毎日1000人を超えるメンバーがウェブにアクセスして、デザインに  
ついて議論している。

スレッドレスコムは小売部門や卸売部門をもっているのに、プラットフォーム  
ではなくて、再販売業者（商業者）の側面をもつ。しかし、それはコミュ  
ニティをもち、メンバーはTシャツのデザインに応募でき、サイト上で批評、  
評価して、メンバー間でフィードバック機能を備え、情報が共有されている。  
これらのTシャツのデザインに関してはマルチサイド・プラットフォームで  
ある。スレッドレスコムは伝統的なサプライチェーン、顧客開拓、広告のよ  
うなマーケティングにまったく依存していない。

### 3.2 イノセントィブコム

伝統的に企業は医薬、バイオテクノロジー、消費財、ハイテク産業では科  
学ベースの研究開発問題をクローズド・システムで行われてきた。これに対  
して、イノセントィブコムはウェブサイト上で、自組織で解決困難な問題  
をもつ探索企業と、その解決者をマッチングするマルチサイド・プラットフォー  
ムである。<sup>(16)</sup> イノセントィブコムは許容可能なソリューションに対して5,000  
ドルから10,000ドルの賞金を提供する。問題解決者は匿名にされている。イ  
ノセントィブコムは、科学的問題をかかえる企業と問題解決者をマッチング  
できるようなシステムを構築する。探索企業は適切なソリューションを選択  
して、賞金を支払い、ソリューションの知的財産権を獲得する。

---

(16) <http://www.innocentive.com/>



イノセンティブは創始者のビンガム (A. Bingham) の科学観に基づいてい<sup>(17)</sup>る。かれによれば、多くの科学的な問題を解決するアプローチがとられ、ソリューションは多様であり、ある問題に最適なソリューションが類似した問題にとって最適とは限らない。しかし、企業内では研究員は少数であるので、問題領域にベストなアプローチやソリューションを気づかないことが多い。企業内部でソリューションが見つからない内部の問題を多様な外部の研究者とマッチングすることが科学的問題解決の生産性を高める。

イノセンティブコムに参加している研究者は世界中で12万人を超える。個別企業やその研究所が解決できない各問題に200人以上の科学者が取り組み、約1/3の問題が解決され、賞金が与えられた。多くのソリューションは予想外の出所から応募され、もともとの問題に取り組んでいた研究者からは生まれなかった。

### 3.3 トップコーダー

トップコーダーはソフトウェア・プロジェクトを仲介している。<sup>(18)</sup>トップコーダーはサイドでソフトウェアの買手と、他方のサイドでは200ヶ国におよぶ約20,000人のデベロッパーをマッチングするマルチサイド・プラットフォームである。それは標準的なソフトウェアの開発で効率を向上させ、プロジェクトの複雑性を節約している。ソフトウェア・デベロッパーとその買手コミュニティは、定期的に予定のコンテストを行って、個々のソフトウェアの問題を解決する。コンテストの勝者は現金の報酬を得ることができる。賞（報酬）は変動する。

このようなシステムは価格メカニズムを利用しているように見えるが、調整ルールが埋め込まれている。価格メカニズムと異なる点は、価格メカニ

---

(17) Lakhani and Panetta [2007].

(18) Boudreau and Hagiu [2009]. <http://www.topcoder.com/>

イノベーションの分散化とプラットフォーム・デザイン（中田善啓）

ムが問題の解決の前に入札が行われるが、トップコーダーは事後的に賞を決定する。したがって、入札よりも競争は激化し、淘汰の役割を果たす。さらに、事後的に賞を与えることは、あるソフトウェア・プロジェクトの最善の解決者に関する不確実性を克服できる。トップコーダーのメンバーは、過去のコンテスト結果に基づく技術評価、提案やパフォーマンスに関する情報を共有している。

### 3.4 コミュニティへ参加する誘因

上述のビジネスモデルでは参加者の報酬は高くはない。また、OSS コミュニティには10万人以上が参加している。そこで、イノベーションの分散化における誘因は明らかにする必要がある。参加者は経済的誘因だけでなく、知的好奇心やアイデンティティやコミュニティの社会的利益に関心を持っている。OSS コミュニティの参加者はタスクを達成感とタスクそれ自身の価値を重視している。さらに、ユーザーがそのニーズから参加する。特に、プログラマーは外部では利用可能でないソフトウェアを必要としているので、開発努力を行う。

OSS コミュニティの参加者は40%が企業からの要請であり、それ以外は仕事以外の目的でそのソフトウェアを利用している。前者は企業が必要とし、それによってコミュニティに貢献する。しかし、OSS コミュニティへの参加報酬は企業の自由裁量で、研究が完成し、評価されて、その参加者と企業に知的財産権が移転されてから支払われる。

このようなピア（仲間、ここでは問題解決者）レビューはソフトウェアの評価であるので、労働市場で評判のシグナルとなる。コミュニティは比較的オープンで透明なプラットフォームとなって、参加者の技能と才能を将来の雇用者にアピールする。雇用者参加者の能力をスクリーニングし、雇用できる。参加者のコミュニティにおける経験は自らの能力を開発し、評判によっ

てキャリアアップに結びつく。このように参加者のチャレンジ、知的好奇心、想像力がOSS、スレッドレスコム、イノセンティブコム、トップコーダーのようなコミュニティはイノベーションの分散化を促進する。

同時に、アイデンティティとコミュニティへの帰属意識は参加に動機づけとなる。コミュニティにおけるアクティブなメンバーはコミュニティの規範にしたがって行動しようとする。たとえば、OSSコミュニティでは参加がオープンであるが、メンバーはフリーで公開し、コードを共有するという規範が受け入れている。多く人が開発したソースコードを利用して利益を得たメンバーはコミュニティに貢献しようとする。

## 4. エコシステムの組織デザイン

### 4.1 コミュニティ

上述のように、コミュニティへの企業や個人の参加によってイノベーションが創発していることは、これまで組織デザインと大きく異なっている。このようなシステムは、伝統的な開発に見られるように、管理者が有能なスタッフをリクルートして、タスクを分割し、同時にそれらを統合して、キャリア・パスなどの誘因を提供して、コントロールされてはいない。コミュニティでの行動をコントロールするのは、相互作用の条件を決定する個々の参加者自身である。

Lakhani and Panetta [2007] はエコシステム内の参加者の運動をブラウン運動ベースのマネジメントと言っている。ブラウン運動は、液体のような溶媒中（媒質としては気体、固体もあり得る）に浮遊する微粒子（たとえばコロイド）が、不規則（ランダム）に運動する現象である。伝統的な組織は参加者が秩序だった行動から成果をうむ。しかし、コミュニティによるイノベーションの分散化は、参加者が自ら参加し、不規則（自らの関心に基づいて）な相互作用からイノベーションが創発することをブラウン運動にたとえてい

る。

OSS コミュニティのケースでは参加者がタスクを自己選抜する。参加者によるタスクの自己選択するので、他のメンバーが取り組んでいる問題やそれから派生する機会から影響を受ける。しかし、そのメンバーが問題解決に取り組むかは自由裁量である。メンバー間に権限関係がないので、あるメンバーが命令を受けることはない。イノセンティブコムやトップコーダーでは、参加者（問題解決者）は問題を自己選択し、自分の知識とソリューションの要件をマッチさせる。スレッドレスコムへ投稿するデザインはメンバーの興味、インスピレーション、グラフィックデザイン能力からうまれる。

## 4.2 知的財産権

伝統的な個別企業の戦略はイノベーションをクローズドにして、顧客をロックイン（囲い込む）し、独占的利益を得ることである。しかし、コミュニティに参加し、イノベーションを促すには、オープン化と透明性が必要である。

これに関連するのは知的財産権である。伝統的なモデルでは、イノベーションを競争企業と顧客から隔離し、多数の知的財産のポートフォリオを蓄積することは、参加者を外部から広くコミュニティへ誘引するには効率的ではない。知的財産をオープン化し、参加者がそれを共有し、知識を再利用するエコシステムが望ましい。

OSS のプロジェクトは著作権やパテントによる知的財産権を保護していない。登録されたコードはその開発者やその企業の著作権が認められるが、参加者は OSS のルールに基づいてそれを利用したり、修正できる。OSS コミュニティは、コミュニティ内外のユーザーが利用できるように、そこで開発したソフトウェアをライセンスしている。フォーマルなライセンス契約はプロジェクト毎に異なるが、OSS のプロジェクトであるグニュー（GNU）プロジェクトが提唱するフリーソフトのウェアのライセンス（GPL）は、ソ

フトウェアがソースプログラムとともにフリーで公開され、使用、複製、変更が可能である。

スレッドレスコムの場合ではデザイナーは、コミュニティ・メンバーにそのデザインを公開して、評価のフィードバックを受け入れなければならない。投稿したアイデアによって他のデザイナーが関連しているが、異なったデザインを創造し、もとのデザインよりも優れていることもある。多数のデザイナーはデザインを公開するが、少数のデザインのみが採用される。

スレッドレスコムは賞を獲得したデザイナーから著作権を得て、それを生産販売することもある。スレッドレスコムは投票ルール、選択ルールをオープンにし、生産についてはコミュニティと相談する。さらには、賞を得たデザインの著作権侵害問題はコミュニティ内で調整されている。このように、スレッドレスコムはマルチサイド・プラットフォームの側面をもっている。

イノセンティブコムは、従来クローズドにされていた内部の科学問題では解決不能な問題を探索する企業とメンバーの問題解決者とマッチングするマルチサイド・プラットフォームである。イノセンティブコムは、探索企業の特定の情報が漏れないようにその企業と問題を一般化する。探索企業と問題解決者は解決プロセスでは匿名にされ、両者の内部の科学的プログラムに関して所有している知識が漏れないようにされている。当該問題に適切なソリューションを見つけた企業は賞と引き替えに、探索企業はその知的財産権を得る。解決者が提供したソリューションを採用されなかった（賞が得られなかった）場合、それは探索企業の知的財産権とならないので、解決者の知的財産権は保護される。

トップコーダーは、デベロッパーと顧客との相互作用を媒介して、ソフトウェアの開発プロセスを調整している。したがって、すべてのコミュニケーション、資産の移転、支払いのような相互作用はプラットフォームを通じて行われる。ソフトウェアの開発とは別に、ソフトウェア・アーキテクチャ、

イノベーションの分散化とプラットフォーム・デザイン（中田善啓）

ソフトウェア・デザイン（仕様、必要条件、モジュールの機能など）、ソフトウェア・アセンブリ（開発されたモジュールを土台にしたアプリケーションの作成）、テスト（ソフトウェアの頑健性を評価のテストのシナリオ）に関するコンテストが開催されている。

アイデアのフリーな移転はしばしば困難な問題に創造的なソリューションを促進する。しかし、その知的財産権問題がコミュニティの拡大を制約する。プラットフォームがオープン性、透明性と継続的なコラボレート参加を維持する知的財産権制度をデザインルールとして設定すれば、イノベーションの分散化が成功する。

## お わ り に

モジュール化とそれによって生まれるモジュールの多様性がイノベーションの分散化を加速する。タスクをモジュール化し、モジュールが多様であれば、多数の参加者がプラットフォームに参加する。スレッドレスコムへの参加者はグラフィックデザイン技能をもつ個人に限定されない。グラフィック・デザイン技能をもたない個人は、デザイナーにアイデアをフィードバックし、デザインの変更を促し、特定のデザインのTシャツのデザインの購入意図を示すので、販売予測の情報を提供している。さらには、そのTシャツを着て、デジタル写真、動画、ブログを提供すること、コミュニティのマーケティングとなる。

コミュニティにおけるタスクをモジュール化は、多様な参加者に問題解決の機会を拡大する。個々の参加者は特定の領域での知識に特化していることが多いので、他の参加者が見逃されていた問題を見つけ、解決することができる。デジタル化、コンピュータ化の情報処理技術のイノベーションはモジュール化のコストを低下させ、同時に多様なモジュールが創出された。ソフトウェアは開発、生産、流通がデジタル化による情報ベースで処理されるので、

コミュニティはイノベーションの分散化の理想的なタイプである。

原理的な科学問題はアカデミックなジャーナルや学会で公表され、研究者間で知識や情報が共有されている。これに対して、イノセントィブコムやトップコーダーは個別の問題を提示し、コンテストベースで実践的なソリューションが提供する。したがって、コンピュータ・シミュレーションは問題解決の有力なツールとなって、ソリューションの実行可能性を近似するので、問題解決の費用が節約される。さらには、ソリューションの解決に必要な実験やシミュレーションの費用は低下しているので、参加者は低費用で参加することができる。

イノベーションの分散化はコミュニティへの自己選択し、参加できるように、プラットフォームをオープン化して、参入障壁がないようなデザインルールを設定しなければならない。モジュール化は、参加者が相互に独立して問題を解決し、再統合するので、大規模な問題をコラボレーションによる解決を促進する。コミュニティによるイノベーションは個々の問題解決者の参加費用が低下すると同時に、幅広い知識と視野が創造的なプロセスで得られるので、イノベーションが加速する。

ジョイの法則に示されているように、コミュニティによる分散的イノベーション・システムは、分布している知識にアクセスして、問題解決するイノベーションのプロセスである。このようなシステムは組織外部の知識を共有することができ、イノベーションを加速する。そのためには、誘因、マネジメント、知的財産権の問題が解決されなければならない。

#### 参 考 文 献

- Amabile, T. and S. Kramer [2011], *The Progress Principle: Using Small Wins to Ignite Joy, Engagement, and Creativity at Work*, Harvard Business Press,
- Baker G., R. Gibbons and K. Murphy (2002) "Relational Contracts and the Theory of the Firm," *The Quarterly Journal of Economics*, 117, pp. 39-84.
- Baldwin, C. Y. [2008], "Where Do Transactions Come from? Modularity, Transactions,

- and the Boundaries of Firms,” *Industrial and Corporate Change*, 17, pp. 155-195.
- Baldwin, C. Y. [2008], “Organization Design for Distributed Innovation,” HBS Working Paper, No.12-100, available at: <http://ssrn.com/abstract=2055814>.
- Baldwin, C. and C. Woodard [2011], “The Architecture of Platforms: A Unified View” in A. Gawer (ed) *Platforms, Markets and Innovation*, Edward Elgar, pp. 19-44.
- Baldwin, C. Y. and E. von Hippel [2010], “Modeling a Paradigm Shift: From Producer Innovation to User and Open Collaborative Innovation,” *Organization Science*, 22, pp. 1399-1417.
- Boudreau, K. and A. Hagiu [2009], “Platform Rules: Multi-sided Platforms as Regulators,” A. Gawer (ed) *Platforms, Markets and Innovation*, Edward Elgar, pp. 163-191
- Boudreau, K. and K. Lakhani, “How to Manage Outside Innovation,” *MIT Sloan Management Review*, pp. 69-76.
- Chen, K. and B. Nalebuff [2006], “One-Way Essential Complements,” Cowles Foundation Discussion Paper No.1588, available at SSRN: <http://ssrn.com.ezpprod1.hul.harvard.edu/abstract=937384>.
- Gawer, A. [2009], “Platform Dynamics and Strategies: From Products to Services,” in A. Gawer (ed.), *Platforms, Markets and Innovation*, E. Elgar, pp. 45-76.
- Hagiu, A. and J. Wright [2011], “Multi-Sided Platform,” HBS Working Paper, 12-024.
- Hart, O. and J. Moore (1990) “Property Rights and the Nature of the Firm,” *Journal of Political Economy*, 98, pp. 1119-1158.
- Lakhani, K. and J. Panetta [2007], “The Principles of Distributed Innovation,” Berkman Center for Internet & Society Research Publication Series, No. 2007-7, available at: <http://ssrn.com/abstract=1021034>.
- 中田善啓 [2009], 『ビジネスモデルのイノベーションプラットフォーム戦略の展開』同文館.
- 中田善啓 [2010], 「イノベーションのガバナンス」 甲南大学経営学会編『経営学の伝統と革新』千倉書房 pp. 291-303.
- 中田善啓 [2011], 「プラットフォームの進化」『甲南経営研究』第51巻, 第1号, pp. 1-31.
- 中田善啓 [2011], 「モジュール化と取引ネットワークのアーキテクチャ」『甲南経営研究』第52巻, 第1号, pp. 115-144.
- 中田善啓 [2012], 「マルチサイド・プラットフォームの再検討」『甲南経営研究』第53巻, 第1号, pp. 1-29.
- von Hippel, E. [2005], *Democratizing Innovation*, MIT Press (サイコム・インターナショナル『民主化するイノベーションの時代』ファーストプレス, 2006年).